players_count

->0, 1, 2, 3, ...**n** ->

How many players a registered to the Race.
-value is used as **player_ID** for user submiting his registration
- value is incremented after each registration
-incremented value will be the next user ID
- Current Value - 1 is latest register user Id

First player is at ID(n) Zero(0) !

Latest n is new player_id

n	= player_id
nR	= player_race_id

{

u64 => String,
0 => "0_user@mail.io",
1 => "1_user@mail.io",
2 => "2_user@mail.io",
3 => "3_user@mail.io",
. . .

players_emails

Stores Email String Text for every player_ID(n) as Key=>Value
n => "example@mail.com"

player_ID => email,
. . .
 n => "n_xyz@mail.io",

}
,

Email with SHA256

{

b256 => struct,
sha256(email) => PlayerProfile,
0xhash_89zx =>{
(u64)level: 0,
(u64)avg_time: 145,
(u64)player_id: **n**,
(u64)total_races: **nR**
},

players_profiles

Stores Player Profile Information by Hashed Email as Key
sha256(email) => PlayerProfile {}

},

n

nR

{

(u64, u64) => struct,
(n, nR) => GameScore,
(1, 4) =>{
(u64)damage: 0,
(u64)finish_time: 145,
(u64)player_id: **n**,
(u64)total_races: **nR**
},

players_game_scores

Stores by Key (Player ID and his Race Number) the value about the
player races results
(player_id, nR) => GameScore {}

},

Example: **players_count n = 3**

players_emails

| ID:**0** => "ana@mail.com" | ID:**1** => "jure@mail.com" | ID:**2** => "luka@mail.com" | ID:**3** => "   " |

players_profiles

anaHASH =>	*jureHASH =>*	*lukaHASH =>*
LEVEL = 10	LEVEL = 0	LEVEL = 6
AVG_TIME = 40	AVG_TIME = 0	AVG_TIME = 25
PLAYER_ID = **0**	PLAYER_ID = **1**	PLAYER_ID = **2**
TOTAL_RACES = **2**	TOTAL_RACES = **0**	TOTAL_RACES = **1**

players_game_scores

GameScore : (**0, 1**)
- DAMAGE: 23
- PLAYER_ID: **0**
- FINSH_TIME: 45
- SEQ_RACE_NUM: **1**

GameScore : (**0, 2**)
- DAMAGE: 34
- PLAYER_ID: **0**
- FINSH_TIME: 35
- SEQ_RACE_NUM: **2**

GameScore : (**2, 1**)
- DAMAGE: 0
- PLAYER_ID: **2**
- FINSH_TIME: 25
- SEQ_RACE_NUM: **1**